(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
9 October 2003 (09.10.2003)

PCT

(10) International Publication Number
WO 03/084181 A1

(51) International Patent Classification⁷: H04L 29/06, 12/26, G06F 1/00

(21) International Application Number: PCT/US03/09665

(22) International Filing Date: 28 March 2003 (28.03.2003)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/319,159 29 March 2002 (29.03.2002) US

(71) Applicant: CISCO TECHNOLOGY, INC. [US/US]; 170 West Tasman Drive, San Jose, CA 95134-1706 (US).

(72) Inventor: ROWLAND, Craig, H.; 6908 Dogwood Hollow, Austin, TX 78750 (US).

(74) Agent: SHOWALTER, Barton, E.; Baker Botts L.L.P., 2001 Ross Avenue, Dallas, TX 75201-2980 (US).

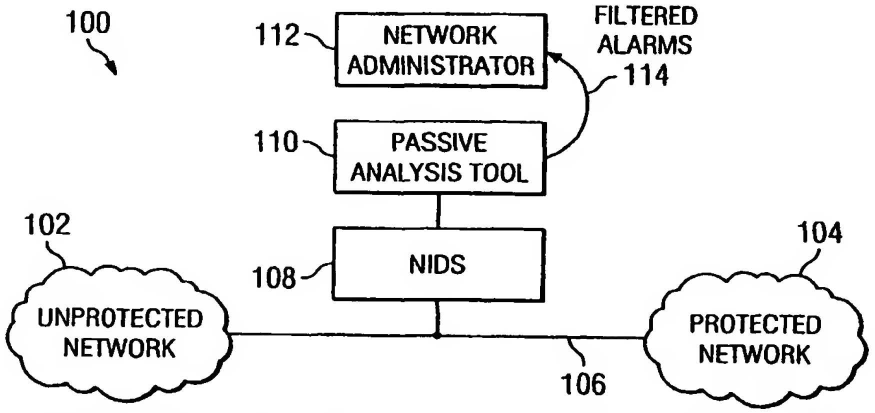
(81) Designated States *(national)*: AE, AG, AL, AM, AT (utility model), AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ (utility model), CZ, DE (utility model), DE, DK (utility model), DK, DM, DZ, EC, EE (utility model), EE, ES, FI (utility model), FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK (utility model), SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, UZ, VC, VN, YU, ZA, ZM, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— *with international search report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: METHOD AND SYSTEM FOR REDUCING THE FALSE ALARM RATE OF NETWORK INTRUSION DETECTION SYSTEMS



(57) Abstract: According to one embodiment of the invention, a method for reducing the false alarm rate of network intrusion detection systems includes receiving an alarm indicating a network intrusion may have occurred, identifying characteristics of the alarm, including at least an attack type and a target address, querying a target host associated with the target address for an operating system fingerprint, receiving the operating system fingerprint that includes the operating system type from the target host, comparing the attack type to the operating system type, and indicating whether the target host is vulnerable to the attack based on the comparison.

METHOD AND SYSTEM FOR REDUCING THE FALSE ALARM RATE OF NETWORK INTRUSION DETECTION SYSTEMS

TECHNICAL FIELD OF THE INVENTION

This invention relates generally to intrusion detection and more particularly to a method and system for reducing the false alarm rate of network intrusion detection systems.

BACKGROUND OF THE INVENTION

Network Intrusion Detection Systems ("NIDS") are typically designed to monitor network activity in real-time to spot suspicious or known malicious activity and to report these findings to the appropriate personnel. By keeping watch on all activity, NIDS have the potential to warn about computer intrusions relatively quickly and allow administrators time to protect or contain intrusions, or allow the NIDS to react and stop the attack automatically. In the security industry, a NIDS may either be a passive observer of the traffic or an active network component that reacts to block attacks in real-time.

Because NIDS are passive observers of the network traffic, they often lack certain knowledge of the attacking and defending host that makes it impossible to determine if an attack is successful or unsuccessful. Much like an eavesdropper overhearing a conversation between two strangers, NIDS very often lack knowledge of the context of the attack and, therefore, "alarm" on network activity that may not be hostile or relevant.

Some systems attempt to address this problem by building a static map of the network they are monitoring. This knowledge is usually built by scanning all the systems on the network and saving the result to a database for later retrieval. This system is inadequate for most networks because the topology, types, and locations of network devices constantly change and requires the administrator to maintain a static database. Additionally, the stress of constantly scanning and keeping the network databases up to date is very intensive and may often slow down or cause network services to stop functioning.

SUMMARY OF THE INVENTION

According to one embodiment of the invention, a method for reducing the false alarm rate of network intrusion detection systems includes receiving an alarm indicating a network intrusion may have occurred, identifying characteristics of the alarm, including at least an attack type and a target address, querying a target host associated with the target address for an operating system fingerprint, receiving the operating system fingerprint that includes the operating system type from the target host, comparing the attack type to the operating system type, and indicating whether the target host is vulnerable to the attack based on the comparison.

Some embodiments of the invention provide numerous technical advantages. Other embodiments may realize some, none, or all of these advantages. For example, according to one embodiment, the false alarm rate of network intrusion detection systems ("NIDS") is substantially reduced or eliminated, which leads to a lower requirement of personnel monitoring of NIDS to respond to every alarm. A lower false alarm rate is facilitated even though knowledge of the entire protected network is not required. Because knowledge of the network is not required, hosts may be dynamically added to the network. According to another embodiment, critical attacks on a network are escalated and costly intrusions are remediated.

Other advantages may be readily ascertainable by those skilled in the art from the following figures, description, and claims.

BRIEF DESCRIPTION OF THE DRAWINGS

For a more complete understanding of the present invention and the advantages thereof, reference is now made to the following description taken in conjunction with the accompanying drawings, wherein like reference numbers represent like parts, and which:

FIGURE 1 is a schematic diagram illustrating a system for reducing the false alarm rate of network intrusion detection systems by utilizing a passive analysis tool according to one embodiment of the invention;

FIGURE 2 is a block diagram illustrating various functional components of the passive analysis tool of FIGURE 1 according to the one embodiment of the invention;

FIGURE 3 is a flowchart illustrating a method for reducing the false alarm rate of network intrusion detection systems according to one embodiment of the invention; and

FIGURE 4 is a flowchart illustrating a method that may be used in conjunction with the method of FIGURE 3 according to one embodiment of the invention.

DETAILED DESCRIPTION OF EXAMPLE EMBODIMENTS OF THE INVENTION

Embodiments of the invention are best understood by referring to FIGURES 1 through 4 of the drawings, like numerals being used for like and corresponding parts of the various drawings.

FIGURE 1 is a schematic diagram illustrating a system 100 for reducing the false alarm rate of a network intrusion detection system ("NIDS") 108 by utilizing a passive analysis tool 110 in accordance with one embodiment of the present invention. In the illustrated embodiment, NIDS 108 is coupled to a link 106 that communicatively couples an unprotected network 102 with a protected network 104. System 100 also includes a network administrator 112 that utilizes passive analysis tool 110, as described in more detail below.

Unprotected network 102 may be any suitable network external to protected network 104. An example of unprotected network 102 is the Internet. Protected network 104 may be any suitable network, such as a local area network, wide area network, virtual private network, or any other suitable network desired to be secure from unprotected network 102. Link 106 couples unprotected network 102 to protected network 104 and may be any suitable communications link or channel. In one embodiment, communications link 106 is operable to transmit data in "packets" between unprotected network 102 and protected network 104; however, communications link 106 may be operable to transmit data in other suitable forms.

In one embodiment, NIDS 108 is any suitable network-based intrusion detection system operable to analyze data packets transmitted over communications link 106 in order to detect any potential attacks on protected network 104. NIDS 108 may be any suitable combination of hardware, firmware, and/or software. Typically, NIDS 108 includes one or more sensors having the ability to monitor any suitable type of network having any suitable data link protocol. In a particular embodiment, the sensors associated with NIDS 108 are operable to examine data packets on an IP ("Internet Protocol") network using any suitable protocol, such as TCP ("Transmission Controlled Protocol"), UDP ("User Datagram Protocol"), and ICMP ("Internet Controlled Message Protocol"). Upon detection of a possible attack on protected network 104, NIDS 108 is operable to generate an alarm indicating that an attack on protected network 104 may have occurred and may block the attack outright. This alarm is then transmitted to passive analysis tool 110 for analysis as described below.

According to the teachings of one embodiment of the present invention, passive analysis tool 110 receives an alarm from NIDS 108 and, using the information associated with the alarm, determines if an attack is real or a false alarm. Passive analysis tool 110 significantly lowers the false alarm rate for network intrusion detection systems, such as NIDS 108, in the network environment and lowers the requirement of personnel, such as network administrator 112, monitoring these systems to respond to every alarm. Details of passive analysis tool 110 are described in greater detail below in conjunction with FIGURES 2 through 4. Although illustrated in FIGURE 1 as being separate from NIDS 108, passive analysis tool may be integral with NIDS 108 such that separate hardware is not required. In any event, NIDS 108 and passive analysis tool 110 work in conjunction with one another to analyze, reduce, or escalate alarms depending on the detected severity and accuracy of the attack. One technical advantage is that the invention may eliminate alarms targeted at the wrong operating system, vendor, application, or network hardware.

Network administrator 112 may be any suitable personnel that utilizes passive analysis tool 110 in order to monitor potential attacks on protected network 104 and respond thereto, if appropriate. Network administrator 112 typically has passive analysis tool 110 residing on his or her computer in order to receive filtered alarms from passive analysis tool, as denoted by reference numeral 114.

FIGURE 2 is a block diagram illustrating various functional components of passive analysis tool 110 in accordance with one embodiment of the present invention. The present invention contemplates more, less, or different components than those shown in FIGURE 2. In the illustrated embodiment, passive analysis tool 110 includes an alarm input layer 202, an alarm interpretation layer 204, a target cache look-up 206, an operating system ("OS") fingerprinting mechanism 208, a port fingerprinting mechanism 210 and an alarm output layer 212. The general functions of each of these components are now described before a more detailed description of the function of passive analysis tool 110 is undertaken in conjunction with FIGURES 3 and 4.

Alarm input layer 202 is generally responsible for accepting the alarm from NIDS 108 and passing it to other system components for analysis. In one embodiment, alarm input layer 202 accepts the alarm from NIDS 108 and determines if the alarm format is valid. If the alarm format is invalid, then the alarm is disregarded. If the alarm format is valid, then the alarm is sent to alarm interpretation layer 204. Alarm input layer 202 is preferably designed to be NIDS vendor independent so that it may accept alarms from multiple NIDS sources concurrently with no modification.

Generally, alarm interpretation layer 204 receives the alarm from alarm input layer 202 and performs an analysis on the alarm. In one embodiment, alarm interpretation layer 204 determines whether the alarm is from a supported NIDS vendor. If the alarm is not from a supported NIDS vendor, an alert is generated and the alarm is disregarded. If the alarm is from a supported NIDS vendor, then alarm interpretation layer 204 is responsible for determining the NIDS vendor alarm type, relevant operating system type being attacked (e.g., Microsoft Windows, Sun Solaris, Linux, UNIX, etc.), the source address, target network address, the alarm severity, the alarm description, and any other suitable parameters associated with the alarm. Some of this information is used by passive analysis 110 to test if the alarm is real or false, as described in more detail below in conjunction with FIGURES 3 and 4.

Target cache look-up 206 indicates that a look-up is performed by passive analysis tool 110 in order to determine if the target host has already been checked for the particular attack indicated by the alarm. The look-up may be performed in any suitable storage location, such as a local state table or database.

OS fingerprinting mechanism 208 performs a passive analysis of the target host to determine the operating system type of the target host. Briefly, in one embodiment, passive analysis tool 110 sends Internet Protocol ("IP") packets at the target host with special combinations of protocol flags, options, and other suitable information in the header in order to ascertain the operating system vendor and version number. Operating system fingerprinting is well known in the industry and, hence, is not described in detail herein. An advantage of this type of OS fingerprinting is that it requires no internal access to the target host other than remote network connectivity. OS fingerprinting mechanism 208 may build an operating system type within seconds of execution and stores this information in a suitable storage location for later retrieval and use.

Port fingerprinting mechanism 210 functions to identify a target port address stored in a suitable storage location when a host is added or deleted dynamically. Port fingerprinting mechanism 210 works in conjunction with OS fingerprinting mechanism 208 to determine, for example, if an attacked port on a target host is active or inactive. This allows passive analysis tool 110 to quickly determine an attack could work. For example, an attack against TCP port 80 on a target host may be proven to have failed by checking the target host to see if port 80 is active to begin with.

Alarm output layer 212 is responsible for taking the analyzed data from passive analysis tool 110 and either escalating or de-escalating the alarm. In other words, alarm output layer 212 functions to report a valid alarm; i.e., that a particular target host is vulnerable to an attack. A valid alarm may be reported in any suitable manner, such as a graphical user interface, a log file, storing in a database, or any other suitable output.

Additional description of the details of the functions of passive analysis tool 110, according to one embodiment of the invention, are described below in conjunction with FIGURES 3 and 4.

FIGURE 3 is a flow chart illustrating an example method for reducing the false alarm rate of network intrusion detection systems according to one embodiment of the present invention. The example method begins at step 300 where an alarm is received from NIDS 108 by passive analysis tool 110. Passive analysis tool 110 identifies the target address from the alarm at step 302. Passive analysis tool 110 then accesses a system cache at step 304 in order to determine if the identified target host has already been checked for that particular attack type.

Accordingly, at decisional step 306, it is determined whether the target address has been found in the system cache. If the target address is found, then at decisional step 308, it is determined whether the cache entry time is still valid. In other words, if a particular target host was checked for a particular type of attack within a recent time period, then this information is stored temporarily in the system cache. Although any suitable time period may be used to store this information, in one embodiment, the information is stored for no more than one hour. If the cache entry time is still valid, then the method continues at step 310 where the OS fingerprint of the target host is received by passive analysis tool 110.

Referring back to decisional steps 306 and 308, if the target address is not found in the system cache or if the cache entry time is invalid for a particular target address that is found in the system cache, then the operating system fingerprint of the target host is obtained by passive analysis tool 110 using any suitable OS fingerprinting technique, as denoted by step 312. The operating system fingerprint is then stored in the system cache at step 314. The method then continues at step 310 where the operating system fingerprint of the target host is received.

The attack type and the operating system type of the target host are compared at step 316 by passive analysis tool 110. At decisional step 318, it is determined whether the operating system type of the target host matches the attack type. If there is a match, then a confirmed alarm is reported by step 320. If there is no match, then a false alarm is indicated, as denoted by step 322. For example, if the attack type is for a Windows system and the operating system fingerprint shows a Windows host, then the alarm is confirmed. However, if the attack type is for a Windows system and the operating system fingerprint shows a UNIX host, then this indicates a false alarm. This then ends the example method outlined in FIGURE 3.

Although the method outlined in FIGURE 3 is described with reference to passive analysis tool 110 comparing an operating system type with an attack type, other suitable characteristics of the operating system may be compared to relevant characteristics of the attack type in order to determine if the alarm is real or false.

Thus, passive analysis tool 110 is intelligent filtering technology that screens out potential false alarms while not requiring knowledge of the entire protected network 104. Alarm inputs are received from a deployed NIDS, such as NIDS 108, and analyzed to determine if an attack is real or a false alarm. This is accomplished even though agents are not required to be installed on each computing device of the protected network 104.

FIGURE 4 is a flowchart illustrating an example method that may be used in conjunction with the example method outlined in FIGURE 3 in accordance with an embodiment of the present invention. The example method in FIGURE 4 begins at step 400 where a dynamic host configuration protocol ("DHCP") server is monitored by passive analysis tool 110. The present invention contemplates any suitable dynamic configuration protocol server being monitored by passive analysis tool 110. At step 402, lease activity is detected by passive analysis tool 110. At decisional step 404 it is determined whether a lease issue is detected or a lease expire is detected.

If a lease expire is detected by passive analysis tool 110, then the system cache is accessed, as denoted by step 406. At decisional step 408, it is determined whether the target address associated with the lease expire is found in the system cache. If the target address is found in the system cache, then the entry is purged, at step 410, from the system cache. Passive analysis tool 110 then continues to monitor the DHCP server. If a target address is not found in the system cache, then the lease expire is disregarded, as denoted by step 412. Passive analysis tool 110 continues to monitor the DHCP server.

Referring back to decisional step 404, if a lease issue has been detected, then the system cache is accessed, as denoted by step 414. At decisional step 416, it is determined whether the target address associated with the lease issue is found in the system cache. If the target address is found, then the entry is purged, at step 418. If the target address is not found in the system cache, then the method continues at step 420, as described below.

At step 420, the operating system fingerprint of a target host is obtained at step 420. The operating system fingerprint is stored in the system cache, as denoted by step 422 for a particular time period. Passive analysis tool 110 then continues to monitor DHCP server.

The method outlined in FIGURE 4 address the dynamic addition of hosts to protected network 104 in order that prior knowledge of the network is not required. This saves considerable time and money and is more accurate than prior systems in which prior knowledge of the network is required. Passive analysis tool 110 may store entries for a user defined length of time that reduces the number of time operating system fingerprints need to be accomplished, which increases the efficiency of the network intrusion detection system. Another technical advantage is that resources are conserved and the impact on the protected network is low because target system profiles are built only when needed, effectively serving as a "just-in-time" vulnerability analysis.

Although the present invention is described with several example embodiments, various changes and modifications may be suggested to one skilled in the art. The present invention intends to encompass those changes and modifications as they fall within the scope of the claims.

WHAT IS CLAIMED IS:

1. A method for reducing the false alarm rate of network intrusion detection systems, comprising:

receiving an alarm indicating a network intrusion may have occurred;

identifying characteristics of the alarm, including at least an attack type and a target address;

querying a target host associated with the target address for an operating system fingerprint;

receiving the operating system fingerprint that includes the operating system type from the target host;

comparing the attack type to the operating system type; and

indicating whether the target host is vulnerable to the attack based on the comparison.

2. The method of Claim 1, further comprising storing the operating system fingerprint of the target host in a storage location for a time period.

3. The method of Claim 1, further comprising:

before querying the target host, accessing a storage location;

determining whether the operating system fingerprint for the target host already exists in the storage location; and

if the operating system fingerprint for the target host does not exist, then continuing the method with the querying step; and

if the operating system fingerprint for the target host does exist, then:

determining if a cache entry time for the target address is valid; and

if the cache entry time is valid, then continuing the method with the comparing step; otherwise

if the cache entry time is invalid, then continuing the method with the querying step.

4. The method of Claim 1, further comprising:

monitoring a dynamic configuration protocol server;

detecting that a lease issue has occurred for a new target host;

accessing a storage location;

determining whether an operating system fingerprint for the new target host already exists in the storage location; and

if the operating system fingerprint for the new target host does not exist, then:

querying the new target host for the operating system fingerprint;

receiving the operating system fingerprint from the new target host; and

storing the operating system fingerprint of the new target host in the storage location for a length of time; and

if the operating system fingerprint for the new target host does exist, then:

purging the existing operating system fingerprint for the new target host from the storage location;

querying the new target host for a new operating system fingerprint;

receiving the new operating system fingerprint from the new target host; and

storing the new operating system fingerprint of the new target host in the storage location for a length of time.

5. The method of Claim 1, further comprising:

monitoring a dynamic configuration protocol server;

detecting that a lease expire has occurred for an existing target host;

accessing a storage location;

determining whether an operating system fingerprint for the existing target host already exists in the storage location; and

if the operating system fingerprint for the existing target host does not exist, then disregarding the lease expire; and

if the operating system fingerprint for the existing target host does exist, then purging the existing operating system fingerprint for the existing target host from the storage location.

6. The method of Claim 1, further comprising:

after receiving the alarm, determining whether a format for the alarm is valid; and

if the format is not valid, then disregarding the alarm; otherwise

if the format is valid, then continuing the method with the identifying step.

7. A method for reducing the false alarm rate of network intrusion detection systems, comprising:

receiving an alarm indicating a network intrusion may have occurred;

identifying characteristics of the alarm, including at least an attack type, a source address, a target address, an alarm severity, and an alarm description;

accessing a storage location;

determining whether an operating system fingerprint for a target host associated with the target address already exists in the storage location;

if the operating system fingerprint for the target host does not exist, then:

querying the target host for the operating system fingerprint;

receiving the operating system fingerprint that includes the operating system type from the target host;

comparing the attack type to the operating system type; and

indicating whether the target host is vulnerable to the attack based on the comparison;

if the operating system fingerprint for the target host does exist, then:

determining if a cache entry time for the target address is valid; and

if the cache entry time is invalid, then:

querying the target host for the operating system fingerprint;

receiving the operating system fingerprint that includes the operating system type from the target host;

comparing the attack type to the operating system type; and

indicating whether the target host is vulnerable to the attack based on the comparison;

if the cache entry time is valid, then:

comparing the attack type to the operating system type; and

indicating whether the target host is vulnerable to the attack based on the comparison.

8. The method of Claim 7, further comprising storing the operating system fingerprint of the target host in the storage location for a time period.

9. The method of Claim 7, further comprising:

monitoring a dynamic configuration protocol server;

detecting that a lease issue has occurred for a new target host;

accessing the storage location;

determining whether an operating system fingerprint for the new target host already exists in the storage location; and

if the operating system fingerprint for the new target host does not exist, then:

querying the new target host for the operating system fingerprint;

receiving the operating system fingerprint from the new target host; and storing the operating system fingerprint of the new target host in the storage location for a length of time; and

if the operating system fingerprint for the new target host does exist, then:

purging the existing operating system fingerprint for the new target host from the storage location;

querying the new target host for a new operating system fingerprint;

receiving the new operating system fingerprint from the new target host; and

storing the new operating system fingerprint of the new target host in the storage location for a length of time.

10. The method of Claim 7, further comprising:

monitoring a dynamic configuration protocol server;

detecting that a lease expire has occurred for an existing target host;

accessing the storage location;

determining whether an operating system fingerprint for the existing target host already exists in the storage location; and

if the operating system fingerprint for the existing target host does not exist, then disregarding the lease expire; and

if the operating system fingerprint for the existing target host does exist, then purging the existing operating system fingerprint for the existing target host from the storage location.

11. A system for reducing the false alarm rate of network intrusion detection systems, comprising:

a network intrusion detection system (NIDS) operable to transmit an alarm indicating an attack on a network may have occurred;

a software program embodied in a computer readable medium, the software program, when executed by a processor, operable to:

receive the alarm;

identify characteristics of the alarm, including at least an attack type and a target address;

query a target host associated with the target address for an operating system fingerprint;

receive the operating system fingerprint that includes the operating system type from the target host;

compare the attack type to the operating system type; and

indicate whether the target host is vulnerable to the attack based on the comparison.

12. The system of Claim 11, further comprising a storage location operable to store the operating system fingerprint of the target host for a time period.

13. The system of Claim 11, wherein the software program is further operable to:

access a storage location before querying the target host;

determine whether the operating system fingerprint for the target host already exists in the storage location; and

if the operating system fingerprint for the target host does not exist, then continue with the querying step; otherwise

if the operating system fingerprint for the target host does exist, then the software program is further operable to:

determine if a cache entry time for the target address is valid; and

if the cache entry time is valid, then continue with the comparing step; otherwise

if the cache entry time is invalid, then continue with the querying step.

14. The system of Claim 11, wherein the software program is further operable to:

monitor a dynamic configuration protocol server;

detect that a lease issue has occurred for a new target host;

access a storage location;

determine whether an operating system fingerprint for the new target host already exists in the storage location; and

if the operating system fingerprint for the new target host does not exist, then the software program is further operable to:

query the new target host for the operating system fingerprint;

receive the operating system fingerprint from the new target host; and store the operating system fingerprint of the new target host in the storage location for a length of time; and

if the operating system fingerprint for the new target host does exist, then the software program is further operable to:

purge the existing operating system fingerprint for the new target host from the storage location;

query the new target host for a new operating system fingerprint;

receive the new operating system fingerprint from the new target host; and

store the new operating system fingerprint of the new target host in the storage location for a length of time.

15. The system of Claim 11, wherein the software program is further operable to:

monitor a dynamic configuration protocol server;

detect that a lease expire has occurred for an existing target host;

access a storage location;

determine whether an operating system fingerprint for the existing target host already exists in the storage location; and

if the operating system fingerprint for the existing target host does not exist, then disregard the lease expire; and

if the operating system fingerprint for the existing target host does exist, then purge the existing operating system fingerprint for the existing target host from the storage location.

16. The system of Claim 11, wherein the software program has no knowledge of the protected network architecture.

17. The system of Claim 11, wherein the NIDS is vendor independent.

18. A system for reducing the false alarm rate of network intrusion detection systems, comprising:

means for receiving an alarm indicating a network intrusion may have occurred;

means for identifying characteristics of the alarm, including at least an attack type and a target address;

means for querying a target host associated with the target address for an operating system fingerprint;

means for receiving the operating system fingerprint that includes the operating system type from the target host;

means for comparing the attack type to the operating system type; and

means for indicating whether the target host is vulnerable to the attack based on the comparison.

19. The system of Claim 18, further comprising means for storing the operating system fingerprint of the target host in a storage location for a time period.

20. The system of Claim 18, further comprising:

means for accessing a storage location;

means for determining whether the operating system fingerprint for the target host already exists in the storage location; and

if the operating system fingerprint for the target host does exist, then means for determining if a cache entry time for the target address is valid.

21. The system of Claim 18, further comprising:

means for monitoring a dynamic configuration protocol server;

means for detecting that a lease issue has occurred for a new target host;

means for accessing a storage location;

means for determining whether an operating system fingerprint for the new target host already exists in the storage location; and

if the operating system fingerprint for the new target host does not exist, then further comprising:

means for querying the new target host for the operating system fingerprint;

means for receiving the operating system fingerprint from the new target host; and

means for storing the operating system fingerprint of the new target host in the storage location for a length of time; and

if the operating system fingerprint for the new target host does exist, then further comprising:

means for purging the existing operating system fingerprint for the new target host from the storage location;

means for querying the new target host for a new operating system fingerprint;

means for receiving the new operating system fingerprint from the new target host; and

means for storing the new operating system fingerprint of the new target host in the storage location for a length of time.

22. The system of Claim 18, further comprising:

means for monitoring a dynamic configuration protocol server;

means for detecting that a lease expire has occurred for an existing target host;

means for accessing a storage location;

means for determining whether an operating system fingerprint for the existing target host already exists in the storage location; and

if the operating system fingerprint for the existing target host does not exist, then disregarding the lease expire; and

if the operating system fingerprint for the existing target host does exist, then means for purging the existing operating system fingerprint for the existing target host from the storage location.

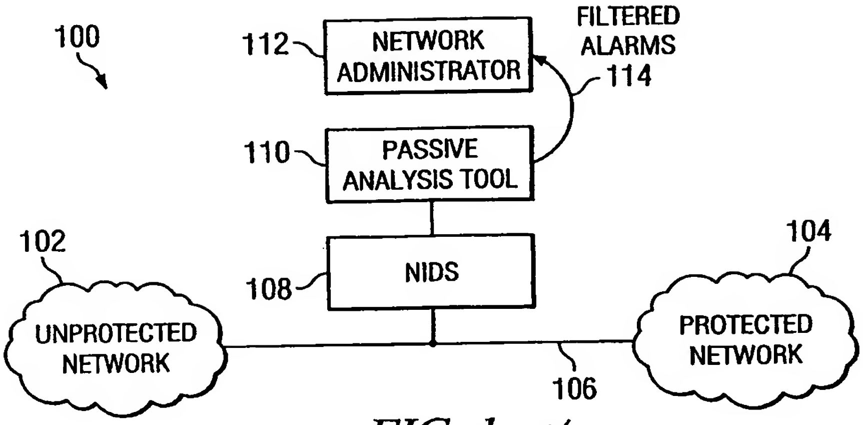

100
112
NETWORK ADMINISTRATOR
FILTERED ALARMS
114
110
PASSIVE ANALYSIS TOOL
102
104
108
NIDS
UNPROTECTED NETWORK
PROTECTED NETWORK
106

FIG. 1

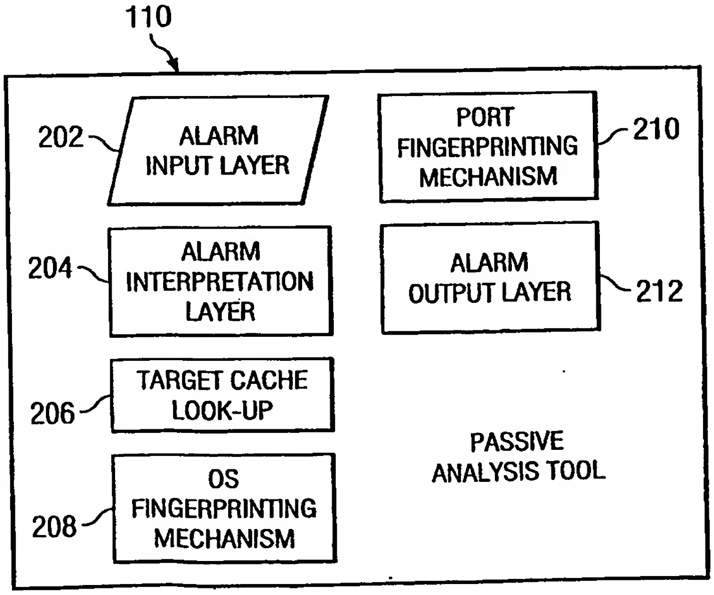

110
202
ALARM INPUT LAYER
PORT FINGERPRINTING MECHANISM
210
204
ALARM INTERPRETATION LAYER
ALARM OUTPUT LAYER
212
206
TARGET CACHE LOOK-UP
208
OS FINGERPRINTING MECHANISM
PASSIVE ANALYSIS TOOL

FIG. 2

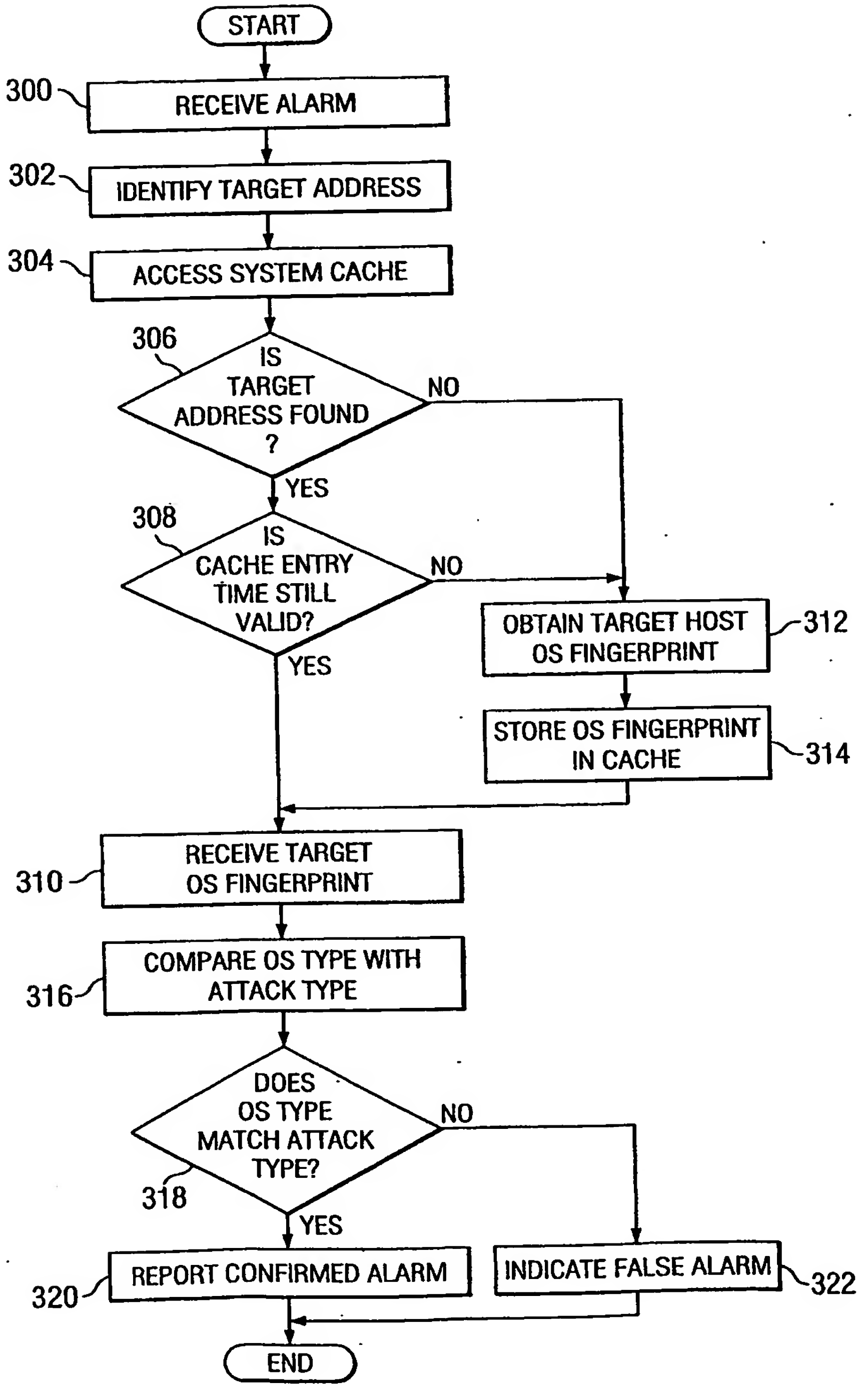

START
300
RECEIVE ALARM
302
IDENTIFY TARGET ADDRESS
304
ACCESS SYSTEM CACHE
306
IS TARGET ADDRESS FOUND ?
NO
YES
308
IS CACHE ENTRY TIME STILL VALID?
NO
YES
OBTAIN TARGET HOST OS FINGERPRINT
312
STORE OS FINGERPRINT IN CACHE
314
RECEIVE TARGET OS FINGERPRINT
310
COMPARE OS TYPE WITH ATTACK TYPE
316
DOES OS TYPE MATCH ATTACK TYPE?
318
NO
YES
REPORT CONFIRMED ALARM
320
INDICATE FALSE ALARM
322
END

FIG. 3

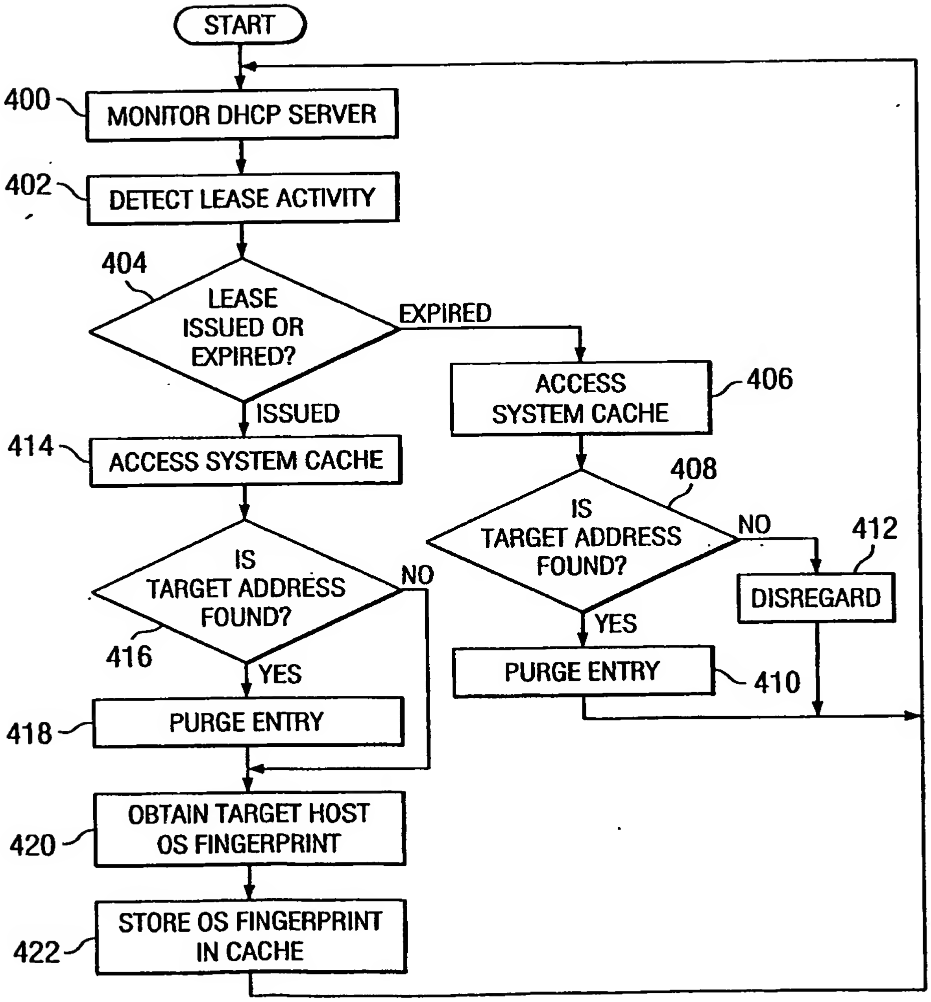

START
400
MONITOR DHCP SERVER
402
DETECT LEASE ACTIVITY
404
LEASE ISSUED OR EXPIRED?
EXPIRED
ISSUED
406
ACCESS SYSTEM CACHE
414
ACCESS SYSTEM CACHE
408
IS TARGET ADDRESS FOUND?
NO
412
DISREGARD
YES
PURGE ENTRY
410
IS TARGET ADDRESS FOUND?
NO
416
YES
418
PURGE ENTRY
420
OBTAIN TARGET HOST OS FINGERPRINT
422
STORE OS FINGERPRINT IN CACHE

FIG. 4

INTERNATIONAL SEARCH REPORT

International Application No
PCT/US 03/09665

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 H04L29/06 H04L12/26 G06F1/00

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 G06F H04L H04Q

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 01 84270 A (INTERNET SECURITY SYSTEMS INC) 8 November 2001 (2001-11-08) page 2, line 12 -page 4, line 12 page 7, line 25 - line 34 abstract; figure 7 ---	1-22
A	EP 0 985 995 A (IBM) 15 March 2000 (2000-03-15) page 3, column 4, line 25 -page 4, column 6, line 7 abstract ---	1-22
A	WO 02 19077 A (JESUS VALDES ALFONSO DE ;SRI INTERNATIONAL INC (US); SKINNER KEITH) 7 March 2002 (2002-03-07) page 2, line 15 -page 3, line 9 abstract; claims 1,2 -----	1-22

□ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search 3 July 2003	Date of mailing of the international search report 21 07 2003
Name and mailing address of the ISA European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Authorized officer RALF BOSTRÖM/JA A

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No
PCT/US 03/09665

Patent document cited in search report		Publication date	Patent family member(s)		Publication date
WO 0184270	A	08-11-2001	AU	5564101 A	12-11-2001
			WO	0184270 A2	08-11-2001
EP 0985995	A	15-03-2000	EP	0985995 A1	15-03-2000
WO 0219077	A	07-03-2002	AU	9501601 A	13-03-2002
			WO	0219077 A2	07-03-2002
			US	2002059078 A1	16-05-2002